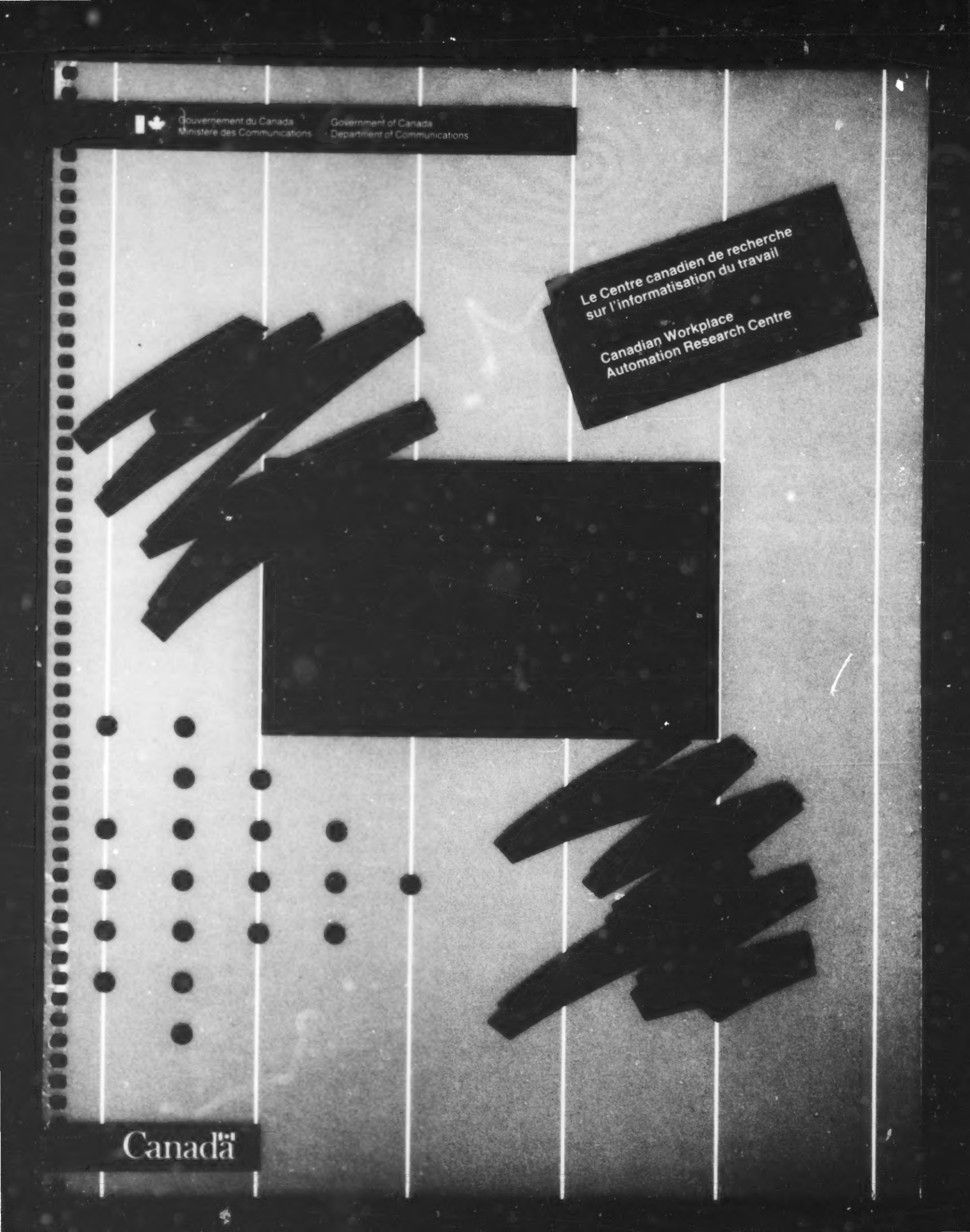
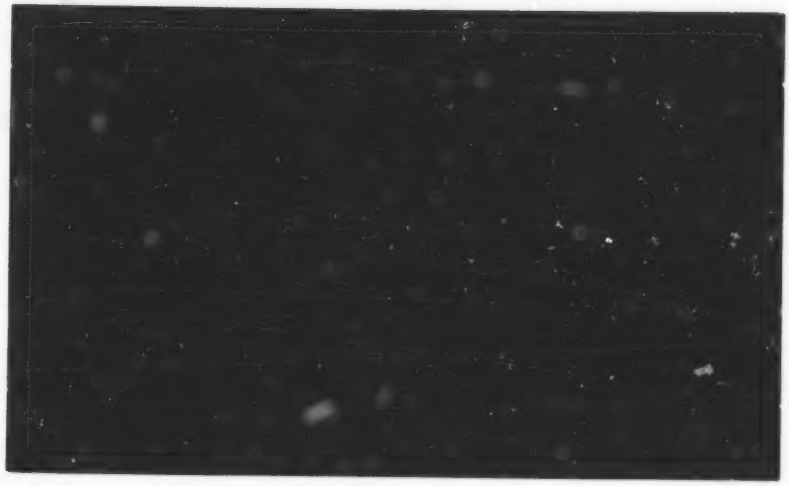

Gouvernement du Canada
Ministère des Communications
Government of Canada
Department of Communications
Le Centre canadien de recherche sur l'informatisation du travail
Canadian Workplace Automation Research Centre
Canada



Transfer and Machine Translation Modularity

by

Pierre Isabelle and Elliott Macklovitch

Researchers
Canadian Workplace Automation Research Centre

This paper was presented at the 11th International Conference on Computational Linguistics, held in Bonn, F.R.G., on August 25-29, 1986.

MCC-CCRIT-86-C06

This article is the fruit of research conducted at the Canadian Workplace Automation Research Centre. The views expressed in this article are strictly those of the authors.

90-06535/

Abstract

The transfer components of typical second generation (G2) MT systems do not fully conform to the principles of G2 modularity, incorporating extensive target language information while failing to separate translation facts from linguistic theory. The exclusion from transfer of all non-contrastive information leads us to a system design in which the three major components operate in parallel rather than in sequence. We also propose that MT systems be designed to allow translators to express their knowledge in natural metalanguage statements.

I. Modularity: a Basic Principle of G2 Systems

Modularity is a defining characteristic of second generation machine translation systems (hereafter G2 MT). G2 systems are claimed to be based on a model in which linguistic descriptions are clearly separated from the algorithms and programs that actually produce translations. Moreover, in this model, the linguistic facts that pertain solely to the source language (SL) are supposed to be clearly separated from the facts that pertain solely to the target language (TL), and from those facts that concern the lexical and structural contrasts between TL and SL. Such, at least, are the principles of G2 design, as set forth for example in Vauquois [1]. This conception of MT gives rise to systems composed of three distinct and successive phases: a monolingual analysis component, which produces a SL-dependent structural description (SD) of the input text; a transfer component, which maps that SD onto a TL-dependent SD; and a monolingual synthesis component, which transforms that SD into a TL output text. (See Figure 1 on the following page.)

As pointed out by Kay [2], this classical G2 design offers a number of advantages. Ideally, it should allow the formal description of a given language to serve the needs of analysis and synthesis indifferently. As well, it should allow a given analysis or synthesis component to be coupled onto other MT systems of similar design to produce translations for other language pairs. And finally, because the algorithms are independent of the particular linguistic descriptions, they too should be reusable in other MT applications.

CLASSICAL G2 DESIGN

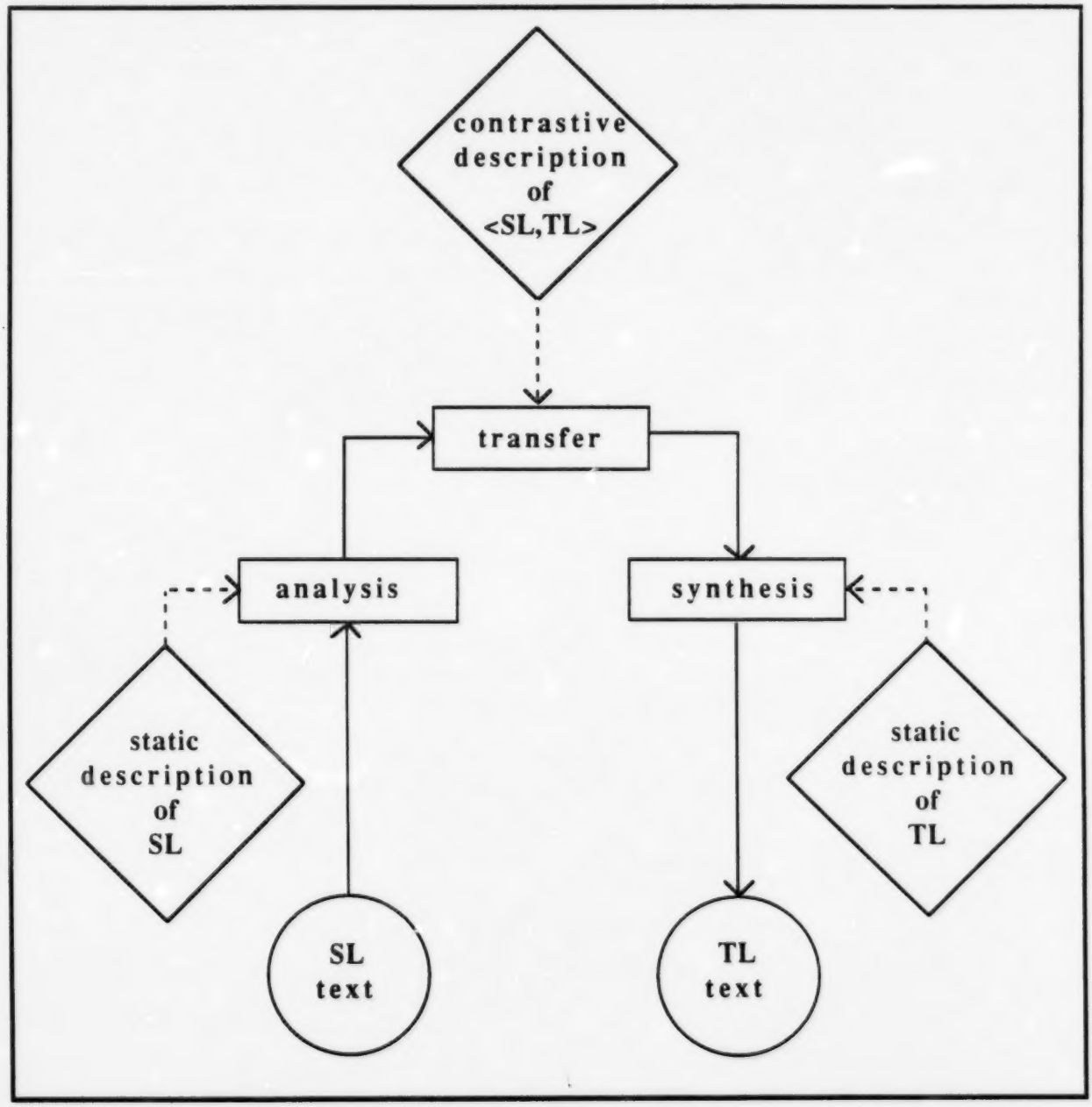
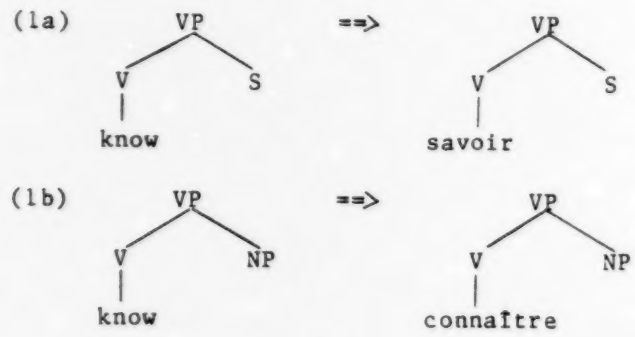


Figure 1

II. Transfer in Typical G2 Systems

Transfer in G2 systems like TAUM-AVIATION [3], ARIANE-78 [4] and METAL [5] is essentially a tree-transformation system, relating the SDs of two complete translation units (generally sentences). Lexical units (LUs) are not translated in isolation; rather, transfer rules typically test the structural environment of each SL LU and, after inserting the appropriate TL equivalent, may rearrange that structure to accord with contextual constraints imposed by the TL LU. Details of formalization aside, transfer rules in these systems will encode facts like the following:



Such transfer rules are usually deterministic: each input tree (or subtree) is mapped onto one and only one output tree. Consequently, corresponding to each SL SD produced by the analysis component, the synthesis component will receive one and only one TL SD. In order for a correct TL sentence to be produced, this unique structure has to be correct in all respects.

This conception makes it necessary for the transfer component to encode a lot of knowledge about TL grammar. For example, if the input tree required by synthesis is an Aspects-type deep structure, transfer will need to include the equivalent of the complete base component of the TL grammar, including lexical insertion mechanisms.

A close examination of the transfer components of the three systems mentioned above shows that this is indeed the case. Rules such as (1a-1b) clearly state facts that belong to a description of TL: namely, strict subcategorization conditions on the insertion of savoir and connaître. A sizable portion of the so-called transfer rules in these systems actually deals with TL strict subcategorization requirements.

In addition to incorporating TL linguistic descriptions within transfer, these systems also make use of impoverished TL grammars for synthesis. For example, none of the systems mentioned accesses a full-fledged TL dictionary during synthesis. It has long been a truism in MT circles that synthesis is much easier that analysis. This is hardly surprising, when so much of the work of synthesis has been passed over to transfer. This move has at least two unfortunate consequences: (a) the burden placed on the translator/lexicographer is greatly increased; and (b) the translation relation is implemented in a highly directional fashion. Analysis and synthesis become totally different tasks, each standing in a different relationship to transfer.

In order to remedy these defects, we favour a more rigourous adherence to the principles of G2 modularity: all information which is not strictly contrastive should be removed from the transfer component and returned to synthesis, where it belongs. This would allow G2 systems to fully benefit from the advantages of modularity that Kay has noted. It would also ease the burden on those who must write the transfer rules.

On the other hand, such a move would make the transfer phase non-deterministic, and this may lead to severe efficiency problems. If, as in typical G2 systems, transfer and synthesis are applied in sequence, communication between the two components will be effected through tree structures representing complete translation units. The use of non-deterministic rules at transfer would mean that local translation ambiguities would then generalize to these complete units. For example, if tranfer no longer selects between savoir and connaître for the translation of know, or between cheveux and poil for the translation of hair, then at least four different French SDs will be transmitted to the synthesis component for any SL SD containing both know and hair. (See Figure 2 on following page.)

One possible solution to this problem is to have analysis, transfer and synthesis operate in parallel rather than in sequence, while providing the three components with the means to exchange partially specified tree structures at each stage of the processing. This is probably what Arnold et al [6] have in mind when they suggest that transfer should be viewed as a relation between two generating devices.

NON-DETERMINISTIC SEQUENTIAL TRANSFER

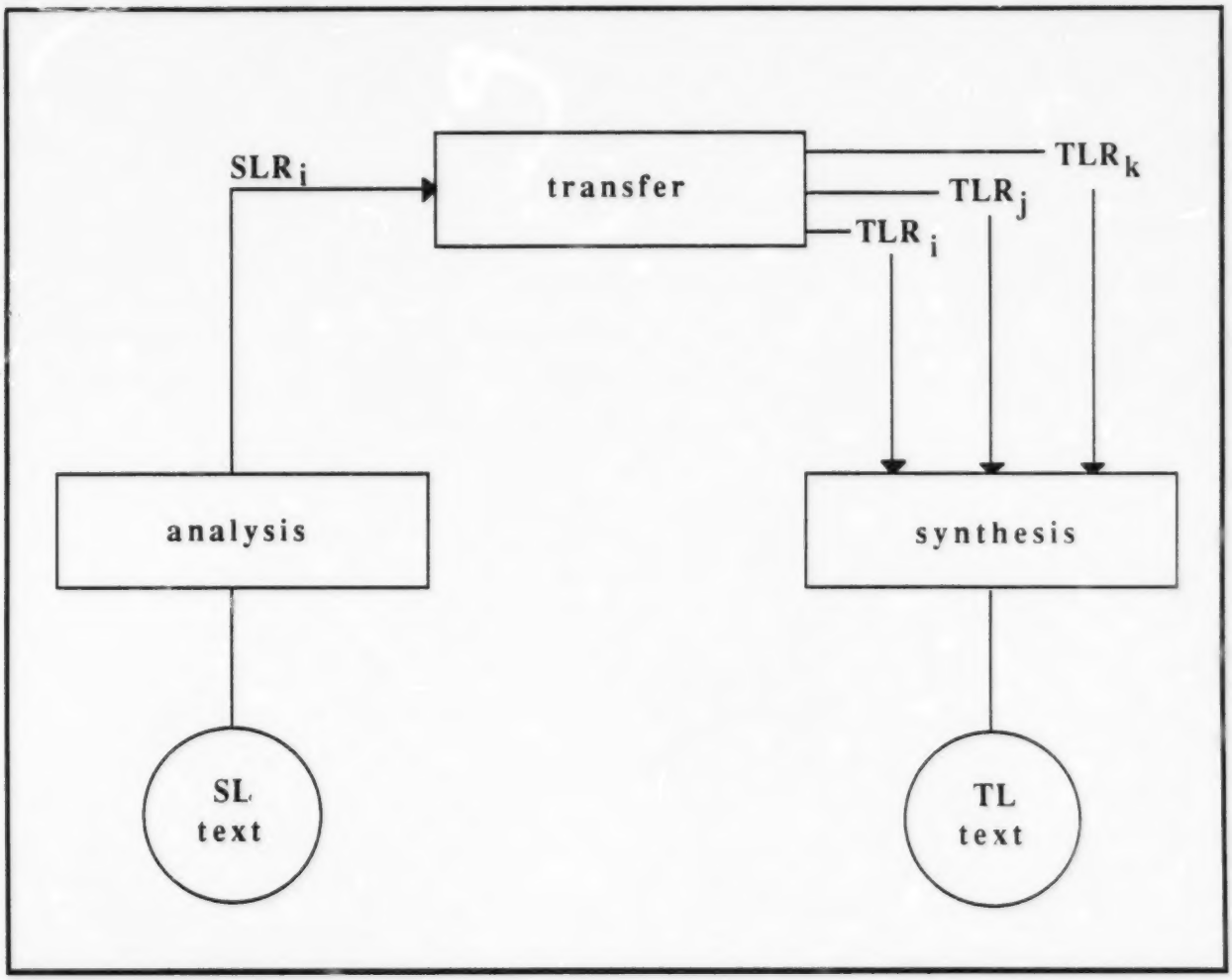


Figure 2

We have recently begun developing a system that seeks to implement such an approach. We have chosen PROLOG II (Université d'Aix-Marseille) as the implementation language, in order to take advantage of the handling of partial information in a time-independent manner that is offered by unification-based formalisms (cf Kay [2]). PROLOG II also has built-in facilities for parallel processing. Though our system is still in the embryonic stage, it can already translate in both directions between English and French, or simply enumerate pairs of sentences that are translationally equivalent.

III. Transfer Rules as Metalanguage Statements

Assuming that the scheme outlined in the preceding paragraphs takes us somewhat closer to the ideals of G2 modularity, we can now ask what content is left for transfer rules to express.

Since a good measure of compositionality is a prerequisite for transfer to be possible at all, the tranfer relation (trf) has to be recursively defined in terms of relations between progressively smaller subparts of the SDs representing complete translation units, up to the level of basic LUs. Depending on the degree of compositionality involved, three classes of rules can be distinguished.

First, there will be rules of purely compositional translation, rules which define the translation of some type of unit as a function of the translation of its parts. Rule (2), for example, states that the translation of an NP is a function of the translation of its components.

```
(2)  trf(np(Det, Adj, N),  np(Det', N', Adj'))
        :-trf(Det, Det')
          trf(Adj, Adj')
          trf(N, N').
```

Second, there will be rules dealing with purely non-compositional cases, such as those of (3):

```
(3)  a) trf(n(chair), n(chaise))
     b) trf(v(know), v(savoir))
        trf(v(know), v(connaître))
     c) trf(nom(n(pressure), n(gauge)),
            nom(n(manomètre))).
```

Example (3a) states a simple correspondence between basic LUs. (3b) illustrates a case where transfer leaves a choice for synthesis, as discussed above. Example (3c) is meant to illustrate the possibility of purely non-compositional translation involving an SL phrase. From the translator's point of view, pressure gauge is a unit; SL-internally, however, we see no reason to consider it an idiom (cf the possibility of conjunction, as in pressure and temperature gauges).

We need a third type of rule to handle partly compositional translation, ie. cases in which the translation of some but not all of the sub-phrases is required for the translation of the whole expression. To

express the restructuring that takes place when miss is translated as manquer à, for example, we would need a rule like (4):

```
(4)  trf(s(Subj,  vp(v(miss), Obj)),
         s(Subj', vp(v(manquer), pp(p(à), Obj'))
        :-trf(Subj, Obj'),
          trf(Obj, Subj').
```

Notice that in the transfer rules given in (2)-(4), just as in those of G2 systems in general, translation facts are cast in terms of the specific linguistic theories embodied in the analysis and synthesis modules. This has a number of serious disadvantages: (a) in order to be able to directly contribute to the development of the transfer component, the translator/lexicographer must assimilate all the details of the specific linguistic theories on which analysis and synthesis are based; (b) descriptions of translation facts have no independent status with respect to the analysis and synthesis modules: changes to the linguistic descriptions of SL and TL will likely force changes to the transfer rules, creating potential consistency problems; and (c) translation rules are unlikely to be portable to other systems.

Natural bilingual dictionaries state translation facts in an altogether different manner. The information they contain is expressed in what essentially amounts to metalanguage statements, like the following:

(5) a) "chair" translates as "chaise" in French
    b) "pressure gauge" translates as "manomètre"
    c) "sb misses sth" translates as "qqch manque à qqn"

Translators obviously manage to express a good deal of translation knowledge in terms of such statements. One way to ease the burden of the translator involved in MT would be to build into the system a capability to automatically interpret statements like those of (5) in terms of rules like those of (3)-(4). In fact, G2 MT systems already possess much of the knowledge required to build SDs out of the quoted expressions in (5). For (5a), little more than SL and TL dictionary lookups are needed to produce (3a). To handle cases such as (5b) the system would only have to access its regular grammar rules. Obviously, other cases will arise where such a simple scheme would lead to the generation of spurious rules. For example, if not supplemented by additional constraints, an equivalence might be created between the English adjective light and the French noun lumière. However, it does not seem difficult to imagine motivated principles that would automatically take care of such incorrect equivalences.

Cases of partly compositional translation raise more interesting problems. In order to produce (4) from (5c), relations must be established between the sub-parts of the two quoted expressions. Sb, sth, qqch and qqn are clearly meant to be interpreted as meta-variables. Sb must be linked to qqn, so that if the first takes some expression as its value, then the second takes the translation of that expression as its value. The clue to this link, of course, is the fact that qqn is itself the translation of sb.

We are currently working on a scheme that will interpret translation rules expressed as simply as those in (5), and allow them to be fed into the transfer component of the MT system mentioned at the end of section II. (See Figure 3 on following page.) Though many difficulties remain to be resolved, the research program we have described does have the merit of addressing the shortcomings that result from the insufficient modularity of G2 systems. At the very least, we hope to stimulate theoretical reflection upon the translation component of MT systems, which has not until recently received the attention it deserves.

INTERPRETING METALANGUAGE STATEMENTS AS TRANSFER RULES

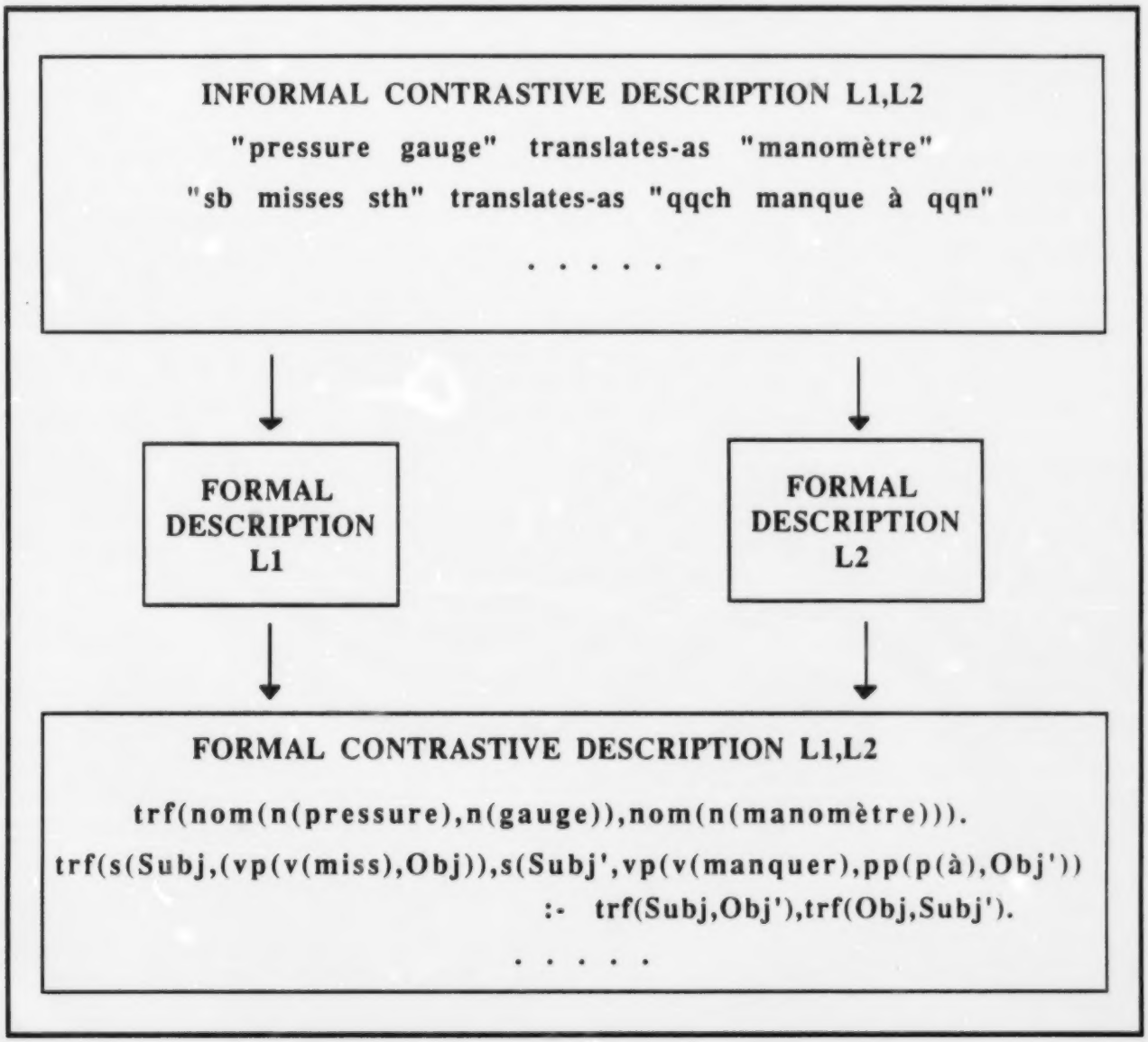


Figure 3

References

Arnold D. & al. (1985). A MU_1 View of the C,A , T Framework in EUROTRA. Proceedings of the Conference on Theoretical Issues in Machine Translation of Natural Languages, Colgate University, 1-14.

Bennett W. & Slocum J. (1985). The LRC Machine Translation System. Computational Linguistics, 11:2-3, 111-121.

Isabelle P. & Bourbeau L. (1985). TAUM-AVIATION: Its Technical Features and some Experimental Results. Computational Linguistics, 11:1, 18-27.

Kay M. (1984). Functional Unification Grammar: a Formalism for Machine Translation. Proceedings of COLING-84, 75-78.

Vauquois, B. (1975). La traduction automatique à Grenoble. Documents de linguistique quantitative, 29, Paris: Dunod.

Vauquois B. & Boitet C. (1985). Automated Translation at Grenoble University. Computational Linguistics, 11:1, 28-36.